

TECHNIQUES FOR INVOKING SYSTEM COMMANDS

FROM WITHIN A MARK-UP LANGUAGE DOCUMENT

1. Field of the Invention

The present invention relates to mark-up language documents, and more
5 particularly to techniques for invoking system commands from within a mark-up
language document.

2. Background of the Invention

Mark-up languages have become exceedingly popular tools for authoring
electronic content. Mark-up languages used today include, for example, Standard
10 Generalized Markup Language (SGML), HyperText Markup Language (HTML),
eXtensible Markup Language (XML), eXtensible HyperText Markup Language
(XHTML), and Synchronized Multimedia Integration Language (SMIL). These
languages allow the developer to "mark-up" a document. For example, tags may be
inserted within the document to specify how the document, or a portion of the
15 document, should be formatted or displayed. As another example, links may be used
to define relationships between the document and other documents and to allow a
user to jump from one document to another. Links may also be used to perform
various functions including, for example, to create tool bars, to link to a cascading
style sheet (rev=stylesheet), a script (rev=script), or a printable version (like a .pdf)
20 of the document (rev=print), and to embed authorship details. Links may also

perform a variety of functions, including executing a string of commands such as JavaScript commands.

When viewed through an application program, such as a web browser or a reading device (discussed herein), links are commonly depicted as underlined text having a blue or red color. In the mark-up language document itself, the link is designated by a link attribute and corresponding language to indicate the resource to which it refers. The link attributes generally include, for example, the following: href (URL for linked resource); id (SGML ID attribute); rel (forward link types); rev (reverse link types); and title (advisory title string). Examples of links within a mark-up language document include:

- 1)
- 2)
- 3)
- 4)

The first link refers to a HyperText Transfer Protocol (HTTP) and provides a Uniform Resource Locator (URL) to which the system should jump. The second link refers to a File Transfer Protocol (FTP) and provides a file the system should retrieve. The third link provides JavaScript that the system should run. The fourth link provides an e-mail address to which the system should send an e-mail message.

Because of their versatility, mark-up languages are now used in a number of applications. One such application is for authoring web-page content. Another such application that is rapidly gaining recognition is for so-called “electronic books” or “e-books.” E-books are electronic publications (such as books, journals, magazines, etc.) that can be viewed using computer-based display devices. Application programs running on these reading devices display the e-book content. Examples of such reading devices include the “ROCKET EBOOK” by NuvoMedia, Inc. and the “SOFTBOOK READER” by Softbook Press, Inc. Alternatively, these devices may be traditional computing devices such as personal computers.

10 sub
A1 E-books are provided in a mark-up language format. As a result, e-book
pages may be formatted and linked using the commands available in the mark-up
language. For example, e-books may be formatted in a general format in accordance
with an Open eBook standard. This standard is set forth in Open eBook Publication
Structure 1.0, which can be found at www.openebook.org. This publication is
15 incorporated herein by reference in its entirety.

One aspect of e-books is that it may be desirable to perform various system commands while viewing an e-book. Consider, for example, a user viewing an e-book from a personal computing environment. The user may wish to adjust the font settings. Present systems are limiting in that the user must take several steps to

invoke system commands and operations. To adjust font settings, the user must exit the window for the e-book viewing program, call up the specific commands from the operating system to adjust the font, and then return to the e-book. This may be particularly disruptive where the user is in the middle of reading an e-book. It is
5 therefore desirable to have a system where system commands may be more conveniently accessed and performed without interrupting the user's activity.

Certain techniques can be used to perform functions within a mark-up language document. In particular, JavaScript may be directly embedded with a tag within the mark-up language document. The tag may designate the JavaScript such
10 that when a user selects the corresponding link, the JavaScript is executed. However, this technique is limiting in that the JavaScript may only perform limited operations and, more significantly, cannot execute system commands outside of the application program reading the JavaScript. Further, the entire JavaScript command set must be included within the mark-up language document. The negative consequence is that
15 possibly undesirable and even malicious functions may be unknowingly executed. Further, the JavaScript-based approach is limiting in that it may unnecessarily increase the size of the document, especially in documents having a large number of JavaScript links. Many e-book documents already are rather large due to the significant amount of text and graphic content. It is therefore desirable to have a

system where pre-defined system commands may be easily identified and executed without requiring increased memory for the document.

SUMMARY OF THE INVENTION

The present invention solves many of the aforementioned problems by
5 providing a method of invoking pre-defined system commands directly from a mark-up language document. Links referencing a pre-defined system command to be invoked may be embedded within the document. The specific system command may be identified in the link by an alias, such as, for example, a numeric code. When a user selects the link, the system will analyze the contents of the link. If the link
10 includes an instruction to invoke a system command, the system will extract the corresponding alias, refer to a look-up table to identify the appropriate system command to which the alias refers, and execute the identified system command. The look-up table correlates each alias with a pre-defined system command.

By using the invoking techniques of the present invention, system commands
15 may be invoked directly from within a mark-up language document. Further, since only those system commands identified in the look-up table can be performed, undesirable or harmful system commands are precluded from being invoked. Even further, desired functionality may be provided in mark-up language documents without the need to enter script in the document itself. Even further, system

commands may be subsequently added, altered, disabled, or removed from the look-up table and the change can thereby be applied to all pre-existing documents. These as well as other novel advantages, details, embodiments, features and objects of the present invention will be apparent to those skilled in the art from following the
5 detailed description of the invention, the attached claims and accompanying drawings, listed herein, which are useful in explaining the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

In the following text and drawings, wherein similar reference numerals denote similar elements throughout the several views thereof, the present invention is
10 explained with reference to illustrative embodiments, in which:

Figure 1 is a schematic block diagram of a computing device for use with the present invention;

Figure 2 is a schematic block diagram of a reading device in accordance with a preferred embodiment of the present invention;

15 Figure 3 is an illustrative table identifying various aliases and the corresponding pre-defined system commands in accordance with the present invention;

Figure 4 is an illustration of the present invention implemented within a reading device for reading e-books;

Figure 5 is a screen print of a page of an e-book having small fonts;

Figure 6 is a screen print of a page of an e-book having large fonts; and

Figure 7 is a flow chart depicting the process by which a system command may be invoked in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

In a preferred embodiment, the present invention is implemented within electronic books (“e-books”) and/or programs and devices for reading e-books. However, the invention may be implemented within, and therefore encompasses, any mark-up language document and any system that interacts with mark-up language documents.

The electronic reading device for use with the present invention may be a separate hand-held computing device or may be an application for use with a more general computing device such as a personal computer (Figure 1). Hand-held reading devices available today include, for example, Windows CE devices.

Figure 1 is a schematic block diagram depicting a personal computing device for use with the present invention. It is envisioned that the hand-held reading device would be a simplified version of personal computer 100. The personal computing device is typically a general-purpose-computing device in the form of a conventional personal computer 100, including a processing unit 110, a system memory 120, and a

system bus 130 that couples various system components including the system memory to the processing unit 110. The system bus 130 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. The system memory includes
5 read only memory (ROM) 140 and random access memory (RAM) 150.

A basic input/output system 160 (BIOS), containing the basic routines that help to transfer information between elements within the personal computer 100, such as during start-up, is stored in ROM 140. The personal computer 100 further includes a hard disk drive 170 for reading from and writing to a hard disk, not
10 shown, a magnetic disk drive 180 for reading from or writing to a removable magnetic disk 190, and an optical disk drive 191 for reading from or writing to a removable optical disk 192 such as a CD ROM or other optical media. The hard disk drive 170, magnetic disk drive 180, and optical disk drive 191 are connected to the system bus 130 by a hard disk drive interface 192, a magnetic disk drive interface
15 193, and an optical disk drive interface 194, respectively. The drives and their associated computer-readable media provide nonvolatile storage of computer readable instructions, data structures, program modules and other data for the personal computer 100.

Although the exemplary environment described herein employs a hard disk, a

removable magnetic disk 190 and a removable optical disk 192, it should be appreciated by those skilled in the art that other types of computer readable media which can store data that is accessible by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, random access
5 memories (RAMs), read only memories (ROMs), and the like, may also be used in the exemplary operating environment. It is anticipated that a hand-held device implementing this invention would typically have only one mass-storage peripheral, either a micro hard disk or else flash memory or equivalent.

A number of program modules may be stored on the hard disk, magnetic disk
10 190, optical disk 192, ROM 140 or RAM 150, including an operating system 195, one or more application programs 196, other program modules 197, and program data 198. A user may enter commands and information into the personal computer 100 through input devices such as a keyboard 101 and pointing device 102. Other input devices (not shown) may include a microphone, joystick, game pad, satellite
15 dish, scanner, or the like. These and other input devices are often connected to the processing unit 110 through a serial port interface 106 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, game port or a universal serial bus (USB). A monitor 107 or other type of display device is also connected to the system bus 130 via an interface, such as a video adapter 108. In

addition to the monitor, personal computers typically include other peripheral output devices (not shown), such as speakers and printers. Preferably, any implementation of the present invention is designed to be operable in a least case scenario only by touch, and does not always require the use of a keyboard or mouse.

5 The personal computer 100 may operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 109. The remote computer 109 may be another personal computer, a server, a router, a network PC, a peer device or other common network node, and typically includes many or all of the elements described above relative to the personal computer 100,
10 although only a memory storage device 111 has been illustrated in Figure 1. The logical connections depicted in Figure 1 include a local area network (LAN) 112 and a wide area network (WAN) 113. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets and the Internet. It is anticipated that a hand-held device used to implement the present invention would
15 typically use a wireless LAN interface based on an infra-red or radio frequency communication link.

When used in a LAN networking environment, the personal computer 100 is connected to the local network 112 through a network interface or adapter 114. When used in a WAN networking environment, the personal computer 100 typically

includes a modem 115 or other means for establishing a communications over the wide area network 113, such as the Internet. The modem 115, which may be internal or external, is connected to the system bus 130 via the serial port interface 106. In a networked environment, program modules depicted relative to the personal computer 100, or portions thereof, may be stored in the remote memory storage device. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be used.

Figure 2 is a simplified schematic block diagram of various elements of a reading device in accordance with a preferred embodiment of the present invention.

10 Resident within memory 120 are a look-up table 300, an application program 196A for reading e-books, one or more e-book files 205, and one or more other application programs 196B (optional). Memory 120 also includes, among other things, an operating system 195. It will be readily apparent that the e-book file may also be resident within other memory devices of the computing system of Figure 1 such as

15 hard disk drive 170, magnetic disk 190, optical disk 192, or remote computer 109. Further, those skilled in the art will appreciate that the present invention may be incorporated in any number of system architectures other than that of Figure 1. In the embodiment of Figure 2, the application program 196A for reading e-books runs on the operating system platform 195, displays one or more e-books 205, and uses

table 300 to identify and invoke pre-defined system commands. Table 300 and the techniques for invoking system commands in accordance with the present invention are discussed herein.

In accordance with a preferred embodiment of the present invention, a link
 5 within a mark-up language document invokes a system command to be performed. System commands are generally functions or operations that may be performed by an operating system or other applications related to the mark-up language document. As preferred, the link is accompanied by an instruction to invoke and an alias to identify a specific system command to be invoked. The alias is a reference code
 10 identifying a pre-defined system command. In this regard, the following link may be used:

`< A HREF=X-MSR1INVOKE: -100>Link Text Goes Here`

The instruction to invoke the system command is the term “X-MSR1INVOKE” and the alias is the term “-100” in the above link. Those skilled in the art will appreciate
 15 that any number of such instructions and aliases may be used and still be considered within the scope of the present invention. The alias refers to a pre-defined system command, which may be ascertained from a look-up table identifying the system commands for each alias. The table 300 of Figure 3 is illustrative in this regard. In the embodiment illustrated in Figure 3, table 300 provides a first column 310 of

various aliases ranging from –100 to +100. A second column 320 identifies the pre-defined system commands corresponding to the alias. For example, the alias –99 identifies the system command for adjusting the contrast a specified amount. Aliases 98-100 remain undefined. The table 300 may be subsequently amended to add new
5 system commands or to alter, disable, and/or remove existing commands.

In addition to aliases having certain pre-defined system commands, certain aliases may also be reserved for future development, including for example, applets for third-party applications. These system commands may be performed, for example, by other application programs 196B.

10 Any number of system commands may be performed by the invoking function of the present invention including, but not limited to, system configuration commands, networked operations, commerce functions, and utilities commands. System configuration commands may include, for example, Brightness (+,-), Contrast (+,-), Page Color (range: 0 to 512), Rotate (90,180,270), Visual Guides
15 (hide/show), All Annotations (hide/show), Bookmarks (hide/show), Highlights (hide/show), Notes (hide/show), Drawings (hide/show), and Riffle Increments (range). Networked operations may include, for example, Network Lookup (lookup a definition of a term in a networked reference source), and Network Search (lookup references to a term in a networked index). Commerce functions may include, for

example, Initiate Transaction, Initiate File Upload, and Form Post (send information to a server). Utilities commands may include, for example, Load Value (entering a value into the system to be operated on by next X-MSR1INVOKE command).

Advantageously, the table 300 pre-defines the various system commands that
 5 may be automatically invoked when the user selects the corresponding link. Further, since only those system commands that are provided in table 300 may be invoked, undesirable or even harmful system commands are thereby precluded from being invoked. Even further, table 300 may be subsequently modified and such changes to the table 300 may be applied to all links referencing alias defined by the table 300.
 10 For example, a system command to increase the font by one increment may be altered to increase the font by three increments. Accordingly, the link referencing the system command to adjust the font will then adjust the font by three increments rather than one.

Figure 4 illustrates an example of how the present invention may be
 15 implemented within a system for reading mark-up language documents. Figure 4 depicts a screen of an application program 400 showing a page from an e-book. The page is essentially a listing of links 405-430 that allow the user to invoke various system commands. The user may click on one of the links 405-430 using a pointing device, such as a mouse 102, thereby automatically executing the selected system

command. In this example, links 405-430 have embedded aliases –1 through 4, respectively, in the mark-up language document. The corresponding pre-defined system commands for these aliases are identified by table 300 of Figure 3. For example, the “Use Large Fonts” link 430 has an alias of 4 and may be written in the

5 mark-up language document as follows:

```
< A HREF=X-MSR1INVOKE: 4>Use Large Fonts</a>
```

By selecting the “Use Large Fonts” link 430, the system command for adjusting font size is automatically invoked and a large font format is thereby applied. The large font setting is preferably applied to all e-books viewed by the

10 reading program 400 until the “Use Small Fonts” link 430 is selected. Alternatively, the system command may apply to only that particular e-book that is being viewed, such that default settings are applied when other e-books are viewed.

Referring still to Figure 4, after the system command has been invoked, link 430 is automatically renamed to “Use Small Fonts”. The system senses that some of

15 the system commands are complements of each other and renames the text visible on the screen to reflect the changed settings. Figures 5 and 6 respectively illustrate the font size of a page of an e-book before and after the link 430 was selected.

In an alternative embodiment, more than one function may be invoked by a link. For example, the following link may be used to invoke two system commands:

< A HREF=X-MSR1INVOKE: -100, -99>Link Text Goes Here

Figure 7 is a flow chart depicting the process by which a system command may be invoked in accordance with a preferred embodiment of the present invention.

Though the present invention covers any mark-up language document and a system
5 that interacts with such mark-up language documents, the following example is
presented in the specific application of a reading device. At step 705, a reading
program parses the mark-up language document to recognize the links that are
present in the document. At step 710, the program monitors whether the user has
selected a link. Once the user selects a link, the program reads the link to determine
10 the form of the link. One option is that the link may identify a standard protocol in
which case the program follows operation in accordance with known techniques
(step 725). For example, if the link refers to another mark-up language document,
the software will navigate to and display that document. The other option is that the
link indicates that a system resource is to be invoked, namely it recites the “X-
15 MSR1INVOKE” protocol. At step 720, the link is then checked to identify the alias
for that link and table 300 is checked to identify the specific system command to be
performed. At step 730, the program performs the pre-defined system command.
The system then returns to step 710 where it waits for the user to select another link.

Advantageously, the present invention overcomes many of the limitation of

the prior art. For example, the present invention provides a technique to invoke system commands automatically from within a mark-up language document. As another example, since only pre-defined system commands may be invoked as provided in look-up table 300, undesirable or even harmful system commands are
5 thereby precluded from being invoked. As yet another example, since the system commands are not embedded within the document itself, the size of the document is reduced. As even another example, table 300 may be subsequently modified to alter existing system commands and even to include additional system commands.

It will be apparent to those skilled in the art that application of the present
10 invention need not merely be utilized in conjunction with a reading device. The present invention may be implemented within any computing environment and may be used with any application using mark-up language documents including, for example and without limitation, a web browser or a word processor.

In the foregoing specification, the present invention has been described with
15 reference to specific exemplary embodiments thereof. Although the invention has been described in terms of a preferred embodiment, those skilled in the art will recognize that various modifications, embodiments or variations of the invention can be practiced within the spirit and scope of the invention as set forth in the appended claims. The specification and drawings are, therefore, to be regarded in an illustrated

B&W 3797.84608 – MS 73285.1

rather than restrictive sense. Accordingly, it is not intended that the invention be limited except as may be necessary in view of the appended claims.

009509 13460
009509 13460